Refine Search

Search Results -

Terms	Documents
L4 and kernel	3

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L5

Refine Search

Recall Text Clear Interrupt

Search History

DATE: Monday, August 22, 2005 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=USPT; PLUR=NO; OP=OR			
L5	L4 and kernel	3	L5
L4	L3 OR l2	13	L4
L3	L1 and retargeting	12	L3
L2	L1 AND retarget	2	L2
L1	717/136,146,162.ccls. OR 703/23-28.ccls. OR 716/18.ccls.	2543	L1

END OF SEARCH HISTORY

Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 3 of 3 returned.

☑ 1. Document ID: US 6760888 B2

L5: Entry 1 of 3 File: USPT Jul 6, 2004

US-PAT-NO: 6760888
DOCUMENT-IDENTIFIER: US 6760888 B2

TITLE: Automated processor generation system for designing a configurable processor and method for the same

DATE-ISSUED: July 6, 2004

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Killian; Earl A.	Los Altos Hills	CA		
Gonzalez; Ricardo E.	Menlo Park	CA		
Dixit; Ashish B.	Mountain View	CA		
Lam; Monica	Menlo Park	CA		
Lichtenstein; Walter D.	Belmont	MA		
Rowen; Christopher	Santa Cruz	CA		
Ruttenberg; John C.	Newton	MA		
Wilson; Robert P.	Palo Alto	CA		
Wang; Albert Ren-Rui	Fremont	CA		
Maydan; Dror Eliezer	Palo Alto	CA		

US-CL-CURRENT: 716/1; 712/1, 712/200, 712/32, 712/35, 712/36, 716/18

ABSTRACT:

An automated processor design tool uses a description of customized processor instruction set extensions in a standardized language to develop a configurable definition of a target instruction set, a Hardware Description Language description of circuitry necessary to implement the instruction set, and development tools such as a compiler, assembler, debugger and simulator which can be used to develop applications for the processor and to verify it. Implementation of the processor circuitry can be optimized for various criteria such as area, power consumption, speed and the like. Once a processor configuration is developed, it can be tested and inputs to the system modified to iteratively optimize the processor implementation. By providing a constrained domain of extensions and optimizations, the process can be automated to a high degree, thereby facilitating fast and reliable development.

8 Claims, 15 Drawing figures
Exemplary Claim Number: 1

Number of Drawing Sheets: 14

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw D

☑ 2. Document ID: US 6477683 B1

L5: Entry 2 of 3 File: USPT Nov 5, 2002

US-PAT-NO: 6477683
DOCUMENT-IDENTIFIER: US 6477683 B1

TITLE: Automated processor generation system for designing a configurable processor and method for the same

DATE-ISSUED: November 5, 2002

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Killian; Earl A.	Los Altos Hills	CA		
Gonzalez; Ricardo E.	Menlo Park	CA		
Dixit; Ashish B.	Mountain View	CA		
Lam; Monica	Menlo Park	CA		
Lichtenstein; Walter D.	Belmont	MA		
Rowen; Christopher	Santa Cruz	CA		
Ruttenberg; John C.	Newton	MA		
Wilson; Robert P.	Palo Alto	CA		
Wang; Albert Ren-Rui	Fremont	CA		
Maydan; Dror Eliezer	Palo Alto	CA		

US-CL-CURRENT: 716/1; 716/18

ABSTRACT:

An automated processor design tool uses a description of customized processor instruction set extensions in a standardized language to develop a configurable definition of a target instruction set, a Hardware Description Language description of circuitry necessary to implement the instruction set, and development tools such as a compiler, assembler, debugger and simulator which can be used to develop applications for the processor and to verify it. Implementation of the processor circuitry can be optimized for various criteria such as area, power consumption, speed and the like. Once a processor configuration is developed, it can be tested and inputs to the system modified to iteratively optimize the processor implementation. By providing a constrained domain of extensions and optimizations, the process can be automated to a high degree, thereby facilitating fast and reliable development.

104 Claims, 15 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 12

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw D

☑ 3. Document ID: US 6219828 B1

L5: Entry 3 of 3 File: USPT Apr 17, 2001

US-PAT-NO: 6219828
DOCUMENT-IDENTIFIER: US 6219828 B1

TITLE: Method for using two copies of open firmware for self debug capability

DATE-ISSUED: April 17, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Lee; Van Hoa	Cedar Park	TX		

US-CL-CURRENT: 717/129; 711/102, 711/103, 711/104, 713/2, 714/100, 714/30, 717/136

ABSTRACT:

A first copy of Open Firmware is loaded into system memory to supply a debug function and a second copy of the same firmware is then loaded to provide functional code which is to be debugged. The first copy of Open Firmware in system memory is designated as the resident debugging function. Kernel code, within the first copy, sets up an executing environment for the debugger, such as system exception handlers and debug console enablement. Normal Open Firmware configuration variables are retrieved from Non-Volatile Random Access Memory ("NVRAM") by the first copy and transmitted to the loader. The second copy of Open Firmware is loaded into system memory to a location specified by the configuration variables. The second copy firmware image is designated as a normal Open Firmware operation in the system. The second copy initially takes over all system exception handlers except instruction breakpoint exception, program interrupt exception and trace exception. The instruction breakpoint exception is utilized to invoke the first copy, or resident debugger, from the normal Open Firmware (second copy) image during code debugging. The two copy debugging configuration is utilized in conjunction with an online machine language assembler and disassembler.

17 Claims, 4 Drawing figures
Exemplary Claim Number: 1
Number of Drawing Sheets: 4

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw D

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Terms	Documents
L4 and kernel	3